CHAPTER 3

Creating Your First Web Page

In This Chapter

- Fire up your editor
- Titling your page
- Identifying the author
- Beginning the BODY
- Adding an apparent title
- Typing and editing text
- Breaking lines and starting new paragraphs
- Creating a list
- Adding a link
- Previewing your page
- Finishing touches
- HTML elements in shorthand

Now that you've gotten a little background on HTML 4 and the importance of separating your content from formatting, it's time to get your hands dirty (figuratively speaking, of course). By the end of this chapter, you'll have a working Web page. We won't make it do tricks yet, but it will have all the basics. Later, we can add the bells and whistles.

Fire Up Your Editor

For this first Web page, you use the plain text editor that came built into your computer. If you are on a PC running Windows 95/98/2000 or NT, from the Start Menu, select Program and then select Accessories and then select Notepad. You will see something similar to Figure 3-1. If you are on a Mac, open SimpleText and you will see something similar to Figure 3-2. If you are in a UNIX environment, open your text editor of choice (vi, pico, jove, or emacs).

Even if you have some fancy tool that was bundled with your computer when you purchased it, or you downloaded from the Web, or you purchased yourself, don't use it yet. There's time for productivity-enhancing tools later. Right now, you need to learn some basic HTML and, depending on the HTML-development program you have, it may get in your way.

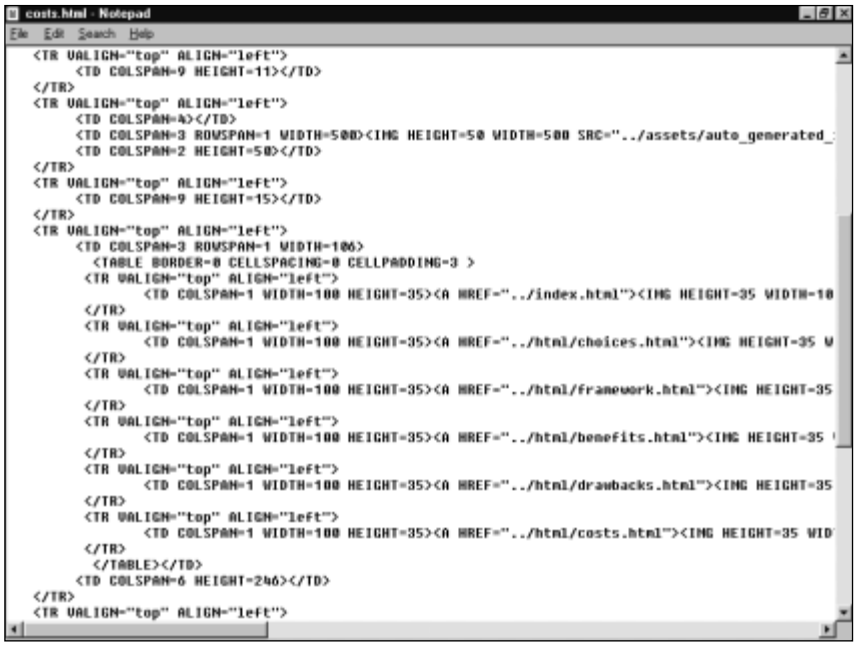


Figure 3-1: Notepad is the text editor that comes with the Windows operating systems.

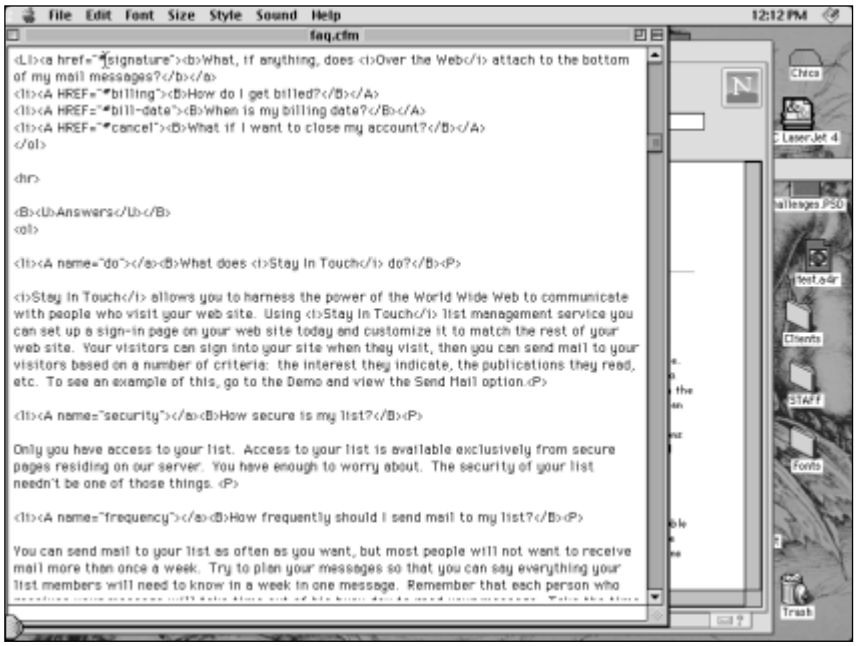


Figure 3-2: SimpleText comes with the Mac operating system.

Titling Your Page

Have you ever tried to set a bookmark for a page, only to find the bookmark had a meaningless name? This was a particular problem in earlier browsers, which didn't permit you to edit bookmark properties. Why didn't a meaningful name — or, sometimes, any name at all — show up in your bookmark list? The Web developer hadn't put a title into the page.

The TITLE element is among the simplest elements. It takes no attributes. In previous versions of HTML, a title was not required. Under HTML 4, absence of a title is a syntax violation. Both a start tag and an end tag are required.

Using as descriptive a title as possible is always a good idea. In addition to showing up in the bookmark folder, the title also displays across the top of your browser. Some Web developers have been in the habit of titling all their pages with the same title — usually the name of the company. This isn't a useful practice because the title should impart useful information. Unlike book titles, where publishers may be imposing limits, with Web page titles, you can take all the words you need to convey your message — within limits, of course. The two major browsers support title length up to 96 or 100 characters. Because search engines frequently use the title of your page as the heading for the entry to your page, it is triply important for you to use a good, descriptive title.

First things first

As you learned in the last chapter, the TITLE element goes in the HEAD element. But where does the HEAD element go? That's right — in the HTML element. So, before you can really put a title on your page, you want to write the following into your empty document:

- Version information
- The HTML element start tag
- The HEAD element start tag

Consider this your first quiz. We give you the answer to the requirement for version information, so you don't have to flip back a few pages, but you must remember what the HTML and HEAD start tags look like.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Strict//EN"
"http://www.w3.org/TR/html40/Strict.dtd">
```

Go ahead and add the HTML element start tag and the HEAD element start tag next. Be sure to put each of these on a separate line. If you want, you can leave blank lines between these tags, as well.

Next, we add the title, which requires the TITLE element start tag: <TITLE>. The content for the TITLE element should be your actual page title, so put that next. At the end of your content, put the TITLE element end tag: </TITLE>.

Identifying the Author

If you recall, we said one other thing went into the HEAD of an HTML document, which is metadata. *Metadata* is information about the information on your page. This would include the name of the author, the software used to write the page, the company name (if there is any), contact information, and so forth.

We won't go into the syntax of the META element right now. You want to indicate you are the author, and there is a META element perfect for this:

```
<META name="Author" content="put your name here">
```

As you can see, the META element has at least two attributes. We have two in the previous example. The META element is unique because, instead of having content between a start tag and an end tag, the META element has a CONTENT attribute. And there is no end tag. You will see a variety of uses for the META element.

In general, you won't see anything from META elements displayed in your browser. The W3C leaves the implementation of the META element to the discretion of the browser. In the future, it is conceivable some of this data could be viewed from within your browser — yet another reason to include it.

Defining key words for better retrieval

While we realize you probably won't try to get this initial Web page listed with a search engine — although you certainly can — you might as well get into using good habits. A good habit to have when creating a Web page is adding a META element for keywords.

Add the following tag to your document:

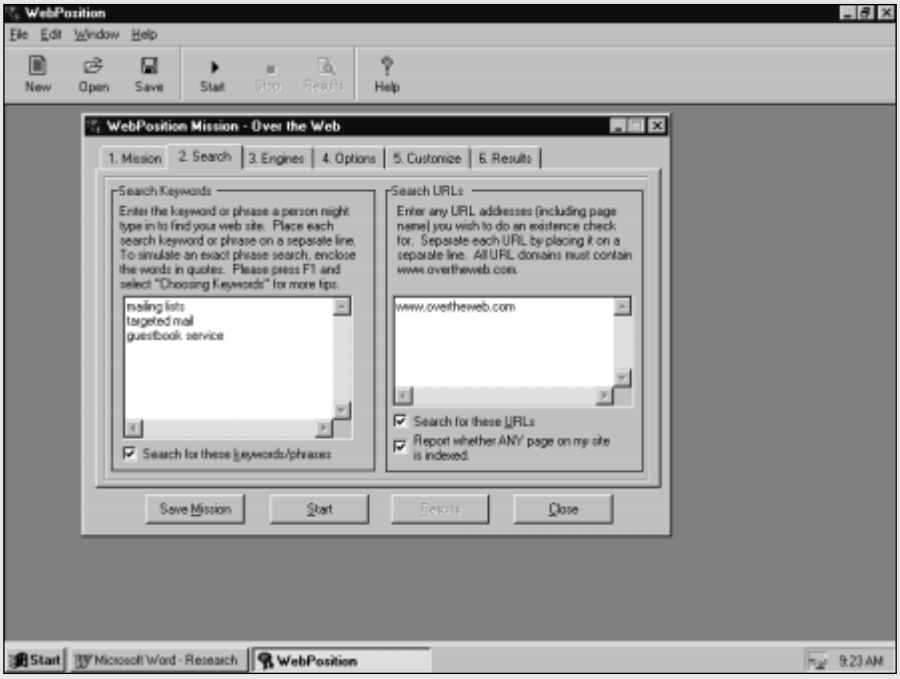
```
<META name="keywords" content="list of keywords">
```

where *list of keywords* is a comma-delimited list such as "list management, email, crossware, relationship management, guestbook, sign-in page." With some search engines, capitalization matters, so it is to your advantage to include *guestbook*, *Guestbook*, and *Guest Book* as three different keywords. Unfortunately, other search engines might interpret this as *spamming* — trying to fool them into listing you higher for a more relevant match.

A note about search engines

Most of the search engines use the metadata you provide about keywords to help index your page. Rules exist about these things, but they change from search engine to search engine. Extensive resources are also on the Web for finding the techniques to get your page listed higher in the search engines. For example, one search engine automatically ignores any word listed more than seven times in the keyword string—even if it is listed in different ways. Knowing the rules when you submit with search engines is important. They are getting harder and harder to fool.

An interesting tool, which can help you identify the ranking of your page on a variety of search engines, is WebPosition Gold. At publication date, this was the only commercial software that could go out to a variety of search engines and find what number your site is, based on the keywords you provide. Many services will do this for you, but WebPosition Gold enables you to do it yourself on a scheduled basis, if you wish, from your own desktop. A trial version of WebPosition Gold is available as a free download at `www.webposition.com/download.htm`. The following figure shows a screen from WebPosition Gold.


WebPosition
File Edit Window Help
New Open Save Start Stop Results Help
WebPosition Mission - Over the Web
1. Mission 2. Search 3. Engines 4. Options 5. Customize 6. Results
Search Keywords
Enter the keyword or phrase a person might type in to find your web site. Place each search keyword or phrase on a separate line. To simulate an exact phrase search, enclose the words in quotes. Please press F1 and select "Choosing Keywords" for more tips.
mailing lists
targeted mail
guestbook service
Search for these keywords/phrases
Search URLs
Enter any URL addresses (including page name) you wish to do an existence check for. Separate each URL by placing it on a separate line. All URL domains must contain www.overtheweb.com.
www.overtheweb.com
Search for these URLs
Report whether ANY page on my site is indexed.
Save Mission Start Results Close
Start Microsoft Word - Research WebPosition 9:23 AM

WebPosition Gold finds your site in all the search engines and reports back to you.

A variety of newsletters can tell you about tips and tricks to get your site listed higher. WebPosition Analyzer publishes its own newsletter. Another excellent one is Danny Sullivan's *Search Engine Report,* which you can subscribe to from the Search Engine Watch page (`www.searchenginewatch.com`).

Beginning the BODY

What do you have to do before you can begin the BODY element? Close the HEAD element. Go ahead and add the end tag for the HEAD element. Then add the start tag for the BODY element.

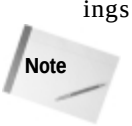
The BODY element has several attributes that you can use to customize the look of your document. In previous versions of HTML, this is where you set the background image, if there was one. This image would tile across the screen, which could look nice or could look horrid. This is also where you would set the background color, the text color, the link color, the visited link color, and the active link color. With the HTML 4 Way, these presentational qualities are all set in the style sheet. This keeps down the amount of markup in your document and keeps the visual clutter to a minimum when you are editing your document. The W3C refers to those BODY attributes as deprecated. A deprecated element or attribute is one the W3C would rather you not use. In future versions of HTML, that element or attribute might not work in browsers or might do something unexpected. For now, all the major browsers support deprecated elements. This means you won't have any problems with them, but it is a good idea to try to steer clear of them as you develop new pages.

Adding an Apparent Title

Going along with these instructions and typing junk every time we instruct you to type content would be easy. Instead, let's stop for a minute. Think about what you want on your page. We are finally at the point where what you type will appear in your browser window.

What did you entitle your page, using the previous TITLE element? Is this really what you want to call your page? If not, go back and change it. Later on in the book, we focus at length on the content of your page, but that doesn't mean you want to publish junk until then.

To add a title to your page, we use the heading elements. In the HTML 4 Way, all we do is tell the browser this is a heading, along with the importance of this heading (number 1 being the most important and number 6 being the least important). As far as presentation goes, we can either let the browser have its way with our headings or we can use style sheets to define the presentation.



Don't be confused by the terms used here. When we discuss the TITLE element, you see the term as uppercase code font. In this discussion of using heading elements on pages, we use title in lowercase.

The heading elements are H1, H2, H3, H4, H5, and H6. Don't try to make them look the way you want your page to look. We can do this later with style sheets. For now, just assign them to headings in order of importance. If you have been in the habit of adding an align attribute, you want to stop doing that. W3C deprecates the use of align as an attribute of heading elements (that's why style sheets are used).

Figure 3-3 shows how you may use elements H1, H2, and H3 to indicate varying degrees of importance in your page.

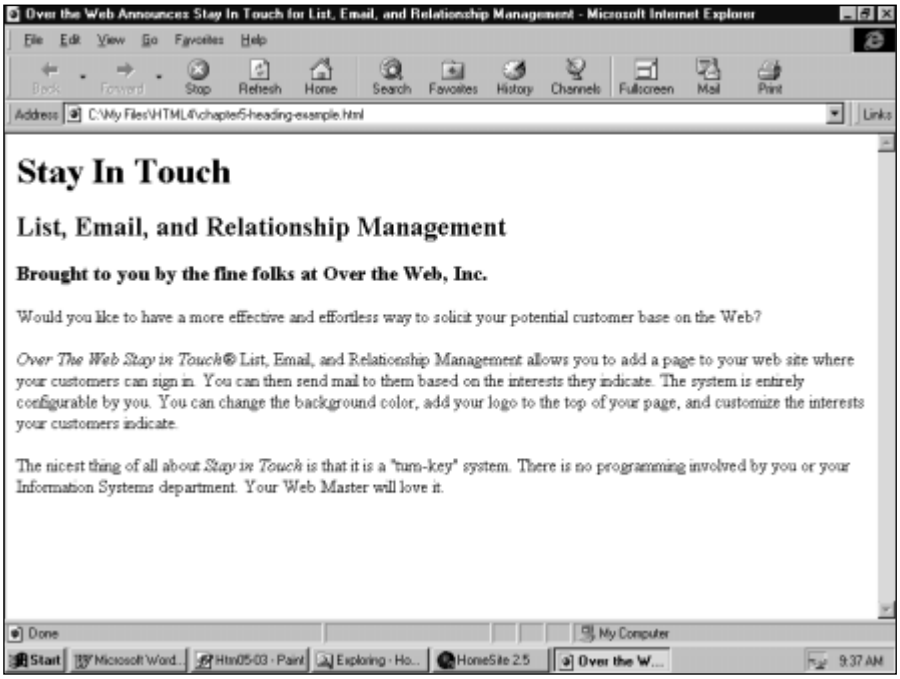

Over the Web Announces Stay In Touch for List, Email, and Relationship Management - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites History Channels Fullscreen Mail Print
Address C:\My Files\HTML4\chapter5-heading-example.html
Links
Stay In Touch
List, Email, and Relationship Management
Brought to you by the fine folks at Over the Web, Inc.
Would you like to have a more effective and effortless way to solicit your potential customer base on the Web?
Over The Web Stay in Touch® List, Email, and Relationship Management allows you to add a page to your web site where your customers can sign in. You can then send mail to them based on the interests they indicate. The system is entirely configurable by you. You can change the background color, add your logo to the top of your page, and customize the interests your customers indicate.
The nicest thing of all about Stay in Touch is that it is a "turn-key" system. There is no programming involved by you or your Information Systems department. Your Web Master will love it.
Done
My Computer
Start
Microsoft Word...
Htm05-03 - Paint
Exploring - Ho...
HomeSite 2.5
Over the W...
9:37 AM

Figure 3-3: Example of using heading elements on your page

This is also a good time to check your own HTML. If the following markup doesn't match your own, you'll want to review your own and see what you missed. Of course, the content won't match.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Strict//EN"
"http://www.w3.org/TR/html40/Strict.dtd">
<HTML>
<HEAD>
```

```
<TITLE>Over the Web Announces Stay In Touch for List, Email,
and Relationship Management</TITLE>
<META name="author" content="Alexis D. Gutzman">
<META name="keywords" content="list management, relationship
management, email, mailing lists, turn-key">
</HEAD>
<BODY>
<h1>Stay In Touch</h1>
<h2>List, Email, and Relationship Management</h2>
<h3>Brought to you by the fine folks at Over the Web, Inc.</h3>
<P>Would you like to have a more effective and effortless way
to solicit your potential customer base on the Web?</P>
<P><I>Over The Web Stay in Touch&reg;</I> List, Email, and
Relationship Management allows you to add a page to your web
site where your customers can sign in. You can then send mail
to them based on the interests they indicate. The system is
entirely configurable by you. You can change the background
color, add your logo to the top of your page, and customize the
interests your customers indicate.</P>
<P>The nicest thing of all about <I>Stay in Touch</I> is that
it is a "turn-key" system. There is no programming involved by
you or your Information Systems department. Your Web Master
will love it.</P>
</BODY>
</HTML>
```

Typing and Editing Text

Now you want to enter your text — the main content of your page. You may want to use two types of text: paragraphs and blockquotes.

Paragraphs

Most prose is divided into paragraphs. HTML provides the `P` element to identify paragraphs. The `P` element has a start tag and an end tag. You put your content between the two. For our purposes in this section, we won't include any attributes. The one you may have seen or used in the past was the `align` attribute, but as we noted, that is deprecated by the W3C.

Blockquotes

The other kind of prose you may want to include is a quote from someone else. If this quote extends beyond a line or two, you will want to set it off in its own block. This is the way we often see reviews of software or reviews of pages. Figure 3-4 shows two quoted paragraphs from Marc Andreessen in a Netscape White Paper:

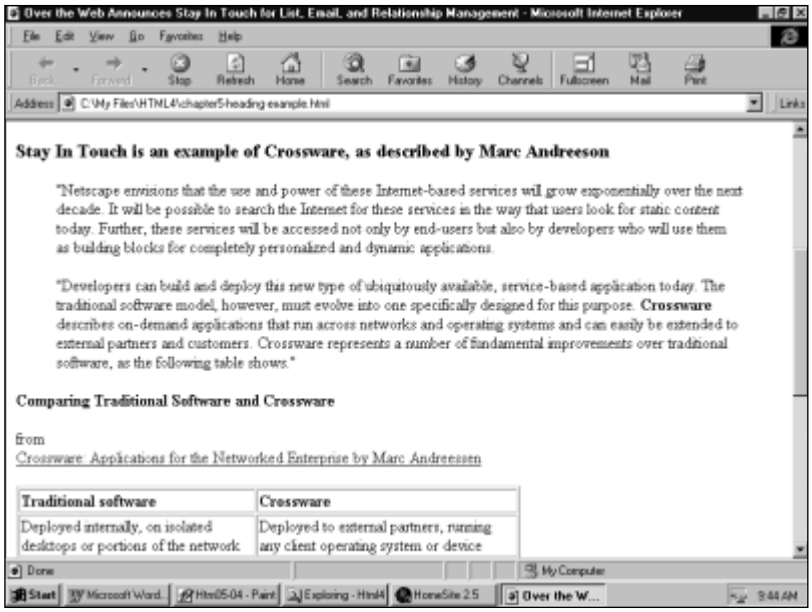

Over the Web Announces Stay In Touch for List, Email, and Relationship Management - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites History Channels Fullscreen Mail Print
Address C:\My Files\HTML4\chapter5-heading example.html
Links
Stay In Touch is an example of Crossware, as described by Marc Andreeson
"Netscape envisions that the use and power of these Internet-based services will grow exponentially over the next decade. It will be possible to search the Internet for these services in the way that users look for static content today. Further, these services will be accessed not only by end-users but also by developers who will use them as building blocks for completely personalized and dynamic applications.
"Developers can build and deploy this new type of ubiquitously available, service-based application today. The traditional software model, however, must evolve into one specifically designed for this purpose. Crossware describes on-demand applications that run across networks and operating systems and can easily be extended to external partners and customers. Crossware represents a number of fundamental improvements over traditional software, as the following table shows."
Comparing Traditional Software and Crossware
from
Crossware: Applications for the Networked Enterprise by Marc Andreessen
Traditional software
Crossware
Deployed internally, on isolated desktops or portions of the network
Deployed to external partners, running any client operating system or device
Done
My Computer
Start
Over the W...
9:44 AM

Figure 3-4: A blockquote

Breaking Lines and Starting New Paragraphs

Sometimes you want to add extra blank space. Perhaps you have text that doesn't belong in a paragraph. In the previous example, after the table heading, we have the word *from* on its own line. To achieve this, we inserted the BR element after the table heading. We wouldn't have wanted this word to be a paragraph by itself, in part, because, structurally, it is not. Had we made this word a paragraph, and in a style sheet indicated that paragraphs should have a half-inch indentation on the first line, then this word would have been indented a half inch. Wouldn't that look strange?

The BR element takes no attributes. It only has a start tag. There is no content and no end tag. This is the easiest element to get right.

```
<H4>The Netscape Application Platform</H4>
<p>from <BR>
<A HREF= "http://developer.netscape.com/docs/wpapers/crossware/index.html">The
Netscape Application Platform by Marc Andreessen</a>
```

Preventing line breaks

The opposite of creating a line break is preventing a line break. Sometimes you want a string of words to appear all on the same line. For example: March 1788. For a variety of reasons, you may always want March and 1788 to appear on the same

line. You can prevent automatic wrapping of lines resulting in March appearing at the end of one line and 1788 appearing at the beginning of the next line by inserting a nonbreaking space between `March` and `1788`. The nonbreaking space is an entity (remember, in entities — unlike the rest of HTML — capitalization *does* matter), but it is not a printable character. To insert a nonbreaking space, you can use either the character representation of ` ` or the decimal representation of ` `. Thus, March 1788 would look like:

```
March 1788
```

Adding horizontal lines

Your page may benefit from horizontal lines. For example, it is not unusual to see a horizontal line between the page and footer text. *Footer text* is text that appears on every page and is not specifically related to the content of the page. For example, you almost always see footers on magazine pages, which may include the page number.

On a Web page, a footer usually includes copyright information, perhaps the date of publication of the page, and how to contact the Webmaster. Many pages include *hit counts*, which actually in no way represent the actual number of visitors, even if they say they do.

Figure 3-5 is an example of a page the effectively uses horizontal lines.

AREA INFO
Thinking about relocating to central Virginia? Already a local? Our area is well-represented on the web, and we've put together this collection of local information to assist you.
The links below will take you to sections of Hyperville--the best guide we've found.
Community Resources
Educational Institutions
Business Directory
Arts and Entertainment
Food and Restaurant Directory
Health and Medical Information
Travel and Tourism
Government Resources
Employment Opportunities
Resources for Senior Citizens
Fun Stuff
ABOUT
AGENTS
LISTINGS
AREA INFO
RESOURCES
All information provided is deemed reliable but is not guaranteed and should be independently verified.
Site design and maintenance by dew Design.
Questions or comments should be directed to the webmaster.

Figure 3-5: The horizontal rule element is right at home at the bottom of this page.

The horizontal rule HR element is almost as foolproof as the BR element. It requires a start tag, forbids an end tag, and has no content. If you have any memories of using attributes with the HR element, forget them. The W3C deprecates any formatting attributes.

Creating a List

You may want to include a list for many reasons:

- ✦ You think in lists.
- ✦ You are trying to persuade people and you want to make your evidence clear.
- ✦ You use lists as shorthand.
- ✦ You want people to know the sequence of events in instructions.
- ✦ You know people more often remember material set apart from the main text.
- ✦ You just want an example of a list.

Whatever your reasons, HTML accommodates your wish for a list. Three types of lists actually exist in HTML:

- ✦ Bulleted lists (called unordered lists)
- ✦ Numbered lists (called ordered lists)
- ✦ Definition lists

We discuss them all in further detail later in the book, but for now we plan to discuss only bulleted lists and numbered lists.

Bulleted lists

We may call them bulleted lists, or simply bullet lists, but HTML calls them *unordered lists*. An unordered list is created with the UL element. For each item in the list, you need the LI element.

The UL element requires a start tag and an end tag. Between those go each item on your list. You can, of course, include other text between the start and end tags, but unless that other text is part of the LI element, it won't appear with a bullet in front of it.

The LI element requires a start tag. The end tag is optional. Following the start tag, you put whatever you want in the list. Each item in your list gets its own LI element. Witness:

```
<UL>
<LI> Have a custom sign-in page on your Website today!
```

```
<LI> Collect information from your customers right on your Web site
<LI> Send email to your customers
<!--LI> Export customer information to spreadsheet, database, or word processing
software for paper mailings-->
<LI> Add your logo, set the background color or graphic to match the rest of
your Web site
</UL>
```

Cross-Reference You can do other interesting tricks with unordered lists such as nesting them, but we save this for Chapter 16, where we cover lists in depth.

Numbered lists

A numbered list uses the `OL` element (ordered list). The `OL` element requires a start tag and an end tag. Each item in the list is part of its own `LI` element. You use the exact same `LI` element as in the ordered list. Look at the following code:

```
<H2>Setting up your list is as Easy as 1-2-3</H2>
<OL>
<LI> Get a Stay in Touch Account
<LI> Click through our easy set-up wizard to customize your pages
<LI> Link to your list from your Web site
</OL>
```

Figure 3-6 shows what the browser does with your lists.

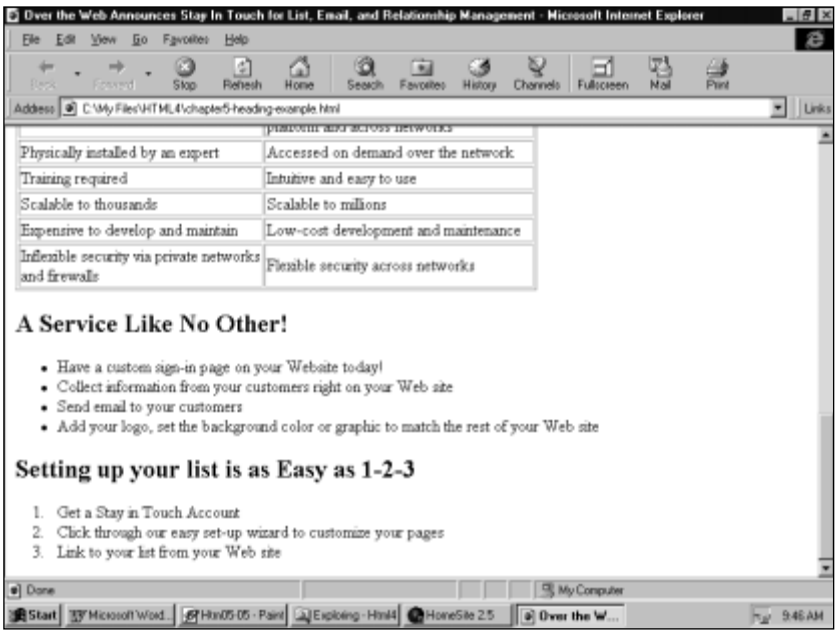

Over the Web Announces Stay In Touch for List, Email, and Relationship Management - Microsoft Internet Explorer
File Edit View Go Favorites Help
Stop Refresh Home Search Favorites History Channels Fullscreen Mail Print
Address C:\My Files\HTML4\chapter5-heading-example.html
Links
Physically installed by an expert
Accessed on demand over the network
Training required
Intuitive and easy to use
Scalable to thousands
Scalable to millions
Expensive to develop and maintain
Low-cost development and maintenance
Inflexible security via private networks and firewalls
Flexible security across networks
A Service Like No Other!
Have a custom sign-in page on your Website today!
Collect information from your customers right on your Web site
Send email to your customers
Add your logo, set the background color or graphic to match the rest of your Web site
Setting up your list is as Easy as 1-2-3
1. Get a Stay in Touch Account
2. Click through our easy set-up wizard to customize your pages
3. Link to your list from your Web site
Done
My Computer
Start
9:46 AM

Figure 3-6: Two kinds of lists

Adding a Link

How can you create a hypertext page without a link — *at least one* link? The syntax for linking is quite simple. You use the A (anchor) element to create a link. Just for fun, let's put one link in your document to something else in your document. The other link, we'll put to the IDG Books Web site. For now, we'll use text to link to the IDG Books Web site, but before you know it, we'll use an image from the CD-ROM in the back of the book. You wondered what was on the CD, didn't you?

Simply, here is the syntax for the A element. Both a start tag and an end tag are required. For the *hot text* (that is, the text that can be clicked to jump elsewhere), content is required. For the destination of the link, content is optional, but we recommend you do include content because some browsers won't find a named anchor (the W3C term for destination) that does not have content.

Adding an internal link

The interesting part of the A element is the attributes. To add an internal link, you use the A element to create both the link and the destination. First, we create the destination. To do this, you need to use the name attribute of the A element, thus:

```
<A name="Features">Features of this product include:</A>
```

Where did we get the value of the name attribute? We made it up. What good does it do to have a destination in our page right now? None. But we can use it in conjunction with hot text elsewhere in the page to make it valuable.

Let's dissect the previous A element, just for review:

Start tag:	`<A name="Features">`
Attribute-value pair:	`name= "Features"`
Attribute:	`name`
Value:	`Features`
Content:	`Features of this product include:`
End tag:	`</A>`

Now, let's link to the named anchor we previously created. We use the href attribute to do this. As with all attributes, the href attribute needs a value. What do you suppose the value of the href attribute will be in our example? If you guess features, you are getting good at this. We just need to add one little twist. We need to add a # (pound sign) in front of the value, inside the double quotation marks.

```
Stay in Touch is loaded with <A href="#Features">
features</A> that no other product has.
```

When you look at this code in a browser, you can't see the named anchor, but if you click the highlighted word *features*, you can jump to the features list. The browser knows where the named anchors are, but it doesn't show them to you.

You've done the harder version; now let's do the easy version.

Adding an external link

When you link to someplace outside your page, whether this is within your site or on some distant server, you use the `A` element with the `href` attribute, such as in the previous example. Pointing to a page somewhere else requires a little more than is required when pointing to a place within your document. The value of the `href` attribute will be a bit longer.

For this example, you are going to point to the IDG Books Worldwide page. You decided that somewhere on your page you mention you're using the HTML 4 Way. This should look familiar to you, with one twist. The value of the `href` attribute is a URL: the home page of IDG Books.

```
<A href="http://www.idgbooks.com">This page was created the HTML 4 Way</A>
```

That's not the most appealing text, but later we'll replace it with an image, anyway. For now, you have a working link to a distant server. Isn't that the point of this whole hypertext thing, anyway?

STOP: Save your file now! Call it MyFirst.html unless you are working in Windows 3.1, in which case, you need to call it MyFirst.htm. If you are working in Windows 3.1, you'll want to remember that every time we talk about your files as something.html, your files will actually be something.htm, okay? This is the last time we'll mention it.

Preview Your Page

Now that you've saved your page, you can open it in a browser and look at it. If anything doesn't look the way you expected, you'll want to review the "Checklist for avoiding common errors when writing HTML" that appears at the end of Chapter 1. Check for end tags (and those pesky slashes). Be sure you have double quotation marks around all your attributes.

You are probably used to opening Web pages in your browser by clicking on the *location* line and typing a URL. The page you have created isn't sitting on a Web browser, so there isn't a URL for it yet. But you can still see your page in your browser. How? On the File Menu of your browser, you'll see either Open Page or Open (depending on the browser maker and version). Select whichever one you have.

Select either Choose File or Browse, depending upon which you have on your screen. Using the standard file management dialog box that opens up, find your HTML file, called MyFirst.html. Select it. Then click Open.

For the remainder of this session, when you make changes to your HTML file, you can just click the Reload or Refresh button at the top of your browser to see your changes.

Finishing Touches

Break out that CD-ROM!

Because you've been good and followed the HTML 4 Way, you're about to be rewarded. We have created four different style sheets to make your page look truly marvelous, darling! This last section tells you where to find the style sheet files, where to copy the style sheet files, and how to link to them.

Loading your style sheets

You could certainly link to style sheets on your CD-ROM without ever copying them, but, because we assume someday you may want to use your CD-ROM drive for something else, we'll tell you how to get them loaded onto your computer.

Copy the four files located in the Chap3 folder on your CD to the same directory where you stored your HTML file called MyFirst.html.

The files are called

```
prof.css
wild.css
retro.css
earthy.css
```

If you want to, you can open them in your text editor and look at them. We tell you what this all means later. If you do open them to read, don't modify them yet. There's time for that later.

Linking to style sheets

Linking to a style sheet requires the use of the `LINK` element. This is a completely different type of link than linking to another page. The `LINK` element takes a start tag, no end tag, no content, and three attributes (when we use it to link to an external style sheet).

The first attribute is the `href` attribute. You should recognize the `href` attribute from the previous `A` element; it does exactly the same thing here. In this case, the value of the `href` attribute is the name of your style sheet. Because all four style sheets are in the same directory where your HTML file is located, you needn't add any directory information in the `href` attribute. We discuss later how you do this when your style sheets are stored elsewhere.

The next attribute is the `rel` (relationship) attribute. This attribute specifies the relationship of this link to your page. The two most common relationship values are *stylesheet,* which means this is the default style sheet for this page, and *alternate stylesheet,* which means this is the alternate style sheet for this page. Because this page only has one style sheet (be patient—we get to the fancy stuff later), you use *stylesheet* as the value of the `rel` attribute.

The final attribute should tell you the `LINK` element has more than one use. In this case, we use the `LINK` element to link to a style sheet, so the value of the `type` attribute is *text/css* (meaning the file is only text and the extension of the file is *css*, which happens to be the extension of the files you copied from the CD-ROM).

This gives us the completed tag:

```
<LINK href="prof.css" rel="stylesheet" type="text/css">
```

You want to put this tag into your `HEAD` element. It can go anywhere in the `HEAD`, but we suggest you put it after your `META` elements. Then you want to save your file and pull it up in your browser. Don't feel you must close out your text editor to see your file or close your browser to edit your file. This will only slow you down. The text editor hardly takes up any resources, anyway.

In case you aren't used to having more than one software program open at a time, you'll find it useful to know, first, that you can have more than one program open and, second, how to switch between them. If you are using any flavor of the Windows operating system, you can switch between open software packages by holding down your Alt key and pressing Tab. You will cycle through all the open software packages (including the Program Manager in Windows 3.1). When you see the icon of the package you want to use, let go of the Alt key. The software you left is still there; it is just in the background.

For a Mac, click the icon in the upper right-hand corner of the screen. You will see the icons of all open software. Click the one you want to use.

Change your style

Now, flip back to your text editor and change the value of the `href` attribute. Use one of the other style sheets. Save your file and look at it. Try them all out and see which one you like best.

Remember, you must first save your HTML file in your editor; then Reload or Refresh your page *before* you can see the changes you made.

The sequence is as follows:

1. Edit your file in your text editor; save your changes.
2. Reload or Refresh your page in your browser.
3. View your changes.

HTML Elements in Shorthand

We intended this chapter to get you up to speed on the basics of marking up your text with elements. If all elements in this book and their attributes were explained in such length as those in this chapter, this book would be three times as thick as it is. This book uses shorthand to give you the basics of each element. This shorthand includes whether there is a start tag, whether there is any content, whether there is an end tag, and whether any of those are forbidden or optional. This shorthand also includes a list of attributes along with information about which, if any, are required, optional, deprecated, or obsolete. We will explain new attributes at length. Attributes you have seen before won't be explained.

Here are two examples:

List Item `<LI>`

Start Tag: required

Content: optional

End Tag: optional

Attributes: none

Blockquote `<BLOCKQUOTE>`

Start Tag: required

Content: required

End Tag: required

Attributes: `cite`: URL

`id`,`class`: document-wide identifiers

`lang`: language information

`dir`: text direction

`title`: element title

`style`: inline style information

From Here

Proceed to Chapter 4 and learn about publishing your page. After all, everyone won't get to see your masterwork until it resides on a Web server.

Already tired of typing all those tags? Jump to Chapter 8 and find the HTML-development software program that's right for you.

Summary

Once you understand the structure of HTML, actually assembling your page is a snap. You want a title, some metadata about your page — those go into the HEAD of your document. Most pages have some headings in the BODY of the page. Add some text, perhaps a blockquote, some lists, and, of course, some links, and you have a solid foundation. You understand the structure of HTML, so actually assembling your page is a snap.

It really gets exciting when you add the style sheets. In one line of code, you can completely change the look of your page. Later, you learn to write your own style sheets, but you can always find some on the Web and link to those (why not?).

✦ ✦ ✦